REMARKS/ARGUMENTS

1. In the above referenced Office Action, the Examiner rejected claims 1-19 under 35 USC § 102 (b) as being anticipated by Everett (U.S. Patent No. 6,220,510). This rejection has been traversed and, as such, the applicant respectfully requests reconsideration of the allowability of claims 1-19.

2. Claims 1-19 have been rejected under 35 USC § 102 (b) as being anticipated by Everett (U.S. Patent No. 6,220,510). The applicant respectfully disagrees with the Examiner's arguments supporting this rejection.

Everett teaches at column 4, lines 20-31, that:

> At a general level, each AAM space created for each application being executed includes two separate address spaces, one for the program code itself and one for the program data which is stored and/or used by the application. The program data address space is effectively divided into three segments: a Static segment, a Dynamic segment and a Public segment which are described in more detail in conjunction with FIG. 1. As stated above, the Static, Dynamic and Public segments are logically mapped to the physical memory; they are virtual memory segments as opposed to physical memory segments. [emphasis added]

Everett further teaches at column 4, lines 53-60, that:

> Within the allocated AAM data space 101, the Static portion of the memory is non-volatile which is not erased after power is removed from the IC card (such as EEPROM), the Dynamic space is volatile (such as RAM) which may be erased after power is removed from the card and the Public space is also volatile (such as RAM). [emphasis added]

Everett further teaches at column 5, lines 49-55, that:

> Referring to FIG. 1, the allocated Static segment 103 contains the application's non-volatile data. Static data includes data which is associated with each application for every transaction such as the card user's name, account number, PIN value and address. Static data also includes variable data which is stored for use in future transactions using the application.

Everett further teaches at column 6, lines 8-12, and column 8, lines 21-23, respectively, that:

> The Dynamic segment 107 contains the application's volatile or temporary data. Dynamic data includes data which is temporarily used during the execution of an application such as intermediate values used in calculations or working variables.
>
> The Public segment 105 is used for command and response data being passed between an IFD and an application.

As such Everett teaches that an AAM creates a virtual memory for program data where data that is to be retained beyond execution of a current application is stored in a static section of the virtual memory (e.g., EEPROM) and data that is not to be retained beyond the execution of a current application is stored a dynamic section of the virtual memory or a public section of the virtual memory. Everett, however, does not teach or suggest allocating a first portion of a first memory as a static section to store a main program which uses functional programs stored in a second memory as is claimed in claim 1. Further, Everett does not teach or suggest allocating a second portion of the first memory as a dynamic section to store other programs, the dynamic section including a plurality of overlay spaces to overlay the functional programs loaded from the second memory to conserve memory capacity of the first memory as is claimed in claim 1.

As such, the applicant contents that claim 1 overcomes the present rejection.

Claims 2-5 are dependent upon claim 1 and introduce additional patentable subject matter. The applicant believes that the reasons that distinguish claim 1 over the present rejection are applicable in distinguishing claims 2-5 over the same rejection.

Claim 6 claims a method that includes executing a program statement of a main program to perform a particular functional operation by identifying a corresponding functional program using a resource identifier and by specifying an entry point into one of the overlay spaces; using the resource identifier to identify a corresponding functional program to perform the particular functional operation; loading the functional program

into an overlay space specified by the specified entry point; and executing the functional program in the overlay space. [emphasis added]

Everett does not teach or suggest such a method. Everett, however, does teach at column 9, lines 12-34, that:

> FIG. 2 shows an extended illustration of the AAM implemented architecture. Data memory space 201 includes the three segments Static, Public and Dynamic as previously described. Code memory space 203 contains the program instructions for an application stored on the IC card. The application instructions are preferably stored in an executable form which can be interpreted by the resident operating system but can also be stored in machine executable form. Instruction 205 is stored at one location in the code memory space 203. Additional instructions are stored in other locations of memory space 203. Two additional registers 207 and 209 are used in the AAM architecture. A code pointer (CP) register 207 indicates the particular code instruction to be next executed. In the figure, the register indicates, e.g., through an offset or pointer means, that instruction 205 is the next to be executed. Condition Control Register 209 contains eight bits, four of which are for use by the individual application and four of which are set or cleared depending upon the results of the execution of an instruction. These condition codes can be used by conditional instructions such as Branch, Call or Jump. The condition codes can include a carry bit, an overflow bit, a negative bit and a zero bit. [emphasis added]

Everett further teaches at column 6, lines 49-51, and at column 10, lines 6-7, respectively, that:

> A delegation function occurs when one application selects another application to process a command instead of processing the command itself.
>
> FIG. 3 shows a flow chart of the steps which are performed when a delegate request is executed.

As such, Everett teaches storing the program instructions for applications in one memory space (e.g., code memory space 203), where one application can delegate a function to be carried out by another application. Everett, however, does not teach loading the functional program into an overlay space specified by the specified entry point; and executing the functional program in the overlay space as is claimed in claim 6. Thus, the applicant believes that claim 6 overcomes the present rejection.

Claims 7-10 are dependent upon claim 6 and introduce additional patentable subject matter. The applicant believes that the reasons that distinguish claim 6 over the present rejection are applicable in distinguishing claims 7-10 over the same rejection.

Claim 11 claims an apparatus that includes a first memory and a second memory. The first memory includes a first portion as a static section to store a main program which uses functional programs and a second portion as a dynamic section to store other programs which reside in the first memory for a shorter duration than the main program, the dynamic section including a plurality of overlay spaces to overlay functional programs. The second memory is operably coupled to store the functional programs and to load a functional program specified by a resource identifier in the main program to a corresponding overlay space specified by an entry point specified by the main program. [emphasis added]

The applicant believes that the arguments that distinguished claim 1 over Everett are applicable in distinguishing claim 11 over Everett.

Claims 12-14 are dependent upon claim 11 and introduce additional patentable subject matter. The applicant believes that the reasons that distinguish claim 11 over the present rejection are applicable in distinguishing claims 12-14 over the same rejection.

Claim 15 claims a multi-function handheld device that includes a system on a chip integrated circuit and an external memory. The system on a chip integrated circuit includes an internal memory arranged to have a first portion as a static section to store a main program which uses functional programs and a second portion as a dynamic section to store other programs which reside in the internal memory for a shorter duration than the main program, the dynamic section including a plurality of overlay spaces to overlay the functional programs. The external memory is operably coupled to the integrated circuit to store the functional programs and to load a functional program specified by a

resource identifier in the main program to a corresponding overlay space specified by an entry point specified by the main program. [emphasis added]

The applicant believes that the arguments that distinguished claim 1 over Everett are applicable in distinguishing claim 15 over Everett.

Claims 16-19 are dependent upon claim 15 and introduce additional patentable subject matter. The applicant believes that the reasons that distinguish claim 15 over the present rejection are applicable in distinguishing claims 16-19 over the same rejection.

For the foregoing reasons, the applicant believes that claims 1-19 are in condition for allowance and respectfully request that they be passed to allowance.

The Examiner is invited to contact the undersigned by telephone or facsimile if the Examiner believes that such a communication would advance the prosecution of the present invention.

RESPECTFULLY SUBMITTED,

By: /Timothy W. Markison reg. 33,534/
Timothy W. Markison
Phone: (808) 665-1725
Fax No. (808) 665-1728

CERTIFICATE OF MAILING
37 C.F.R 1.8

I hereby certify that this correspondence is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner of Patents and Trademarks, Alexandria, Virginia 22313, on the date below:

2-15-06 Diane Hudson

Date Signature